

SurfCuit: Surface Mounted Circuits on 3D Prints

Nobuyuki Umetani

Ryan Schmidt

Autodesk Research

{nobuyuki.umatani,ryan.schmidt}@autodesk.com

ABSTRACT

We present, SurfCuit, a novel approach to design and construction of electric circuits on the surface of 3D prints. Our surface mounting technique allows durable construction of circuits on the surface of 3D prints. SurfCuit does not require tedious circuit casing design or expensive set-ups, thus we can expedite the process of circuit construction for 3D models. Our technique allows the user to construct complex circuits for consumer-level desktop fused decomposition modeling (FDM) 3D printers. The key idea behind our technique is that FDM plastic forms a strong bond with metal when it is melted. This observation enables construction of a robust circuit traces using copper tape and soldering. We also present an interactive tool to design such circuits on arbitrary 3D geometry. We demonstrate the effectiveness of our approach through various actual construction examples.

Author Keywords

Creativity support tools; DIY; Fabrication; Rapid prototyping; 3D Printing; Electronics; Physical computing

ACM Classification Keywords

H.5.m. Information Interfaces and Presentation (e.g. HCI): Miscellaneous

INTRODUCTION

Recent advances in consumer 3D printing technology have made it possible for end users to casually fabricate 3D plastic objects. Many 'makers' would like to add interactivity to their printed objects using sensors, lights, motors, and so on. However, incorporating the necessary electric circuits into these objects has not become inherently easier with 3D printing. Electric circuits are typically designed in 2D, and mounted on planar geometries, such as printed circuit boards (PCBs). Inserting a flat circuit inside a 3D object requires extensive geometry editing to create cavities, wire routing paths, and fixtures. This is generally beyond the reach of the novice or casual maker.

An alternative to is to use 3D circuitry, where 3D traces are embedded into the object volume or surface. However existing CAD interfaces and fabrication techniques have not been designed with 3D circuits in mind. In this paper, we demonstrate both a design tool and fabrication technique to integrate

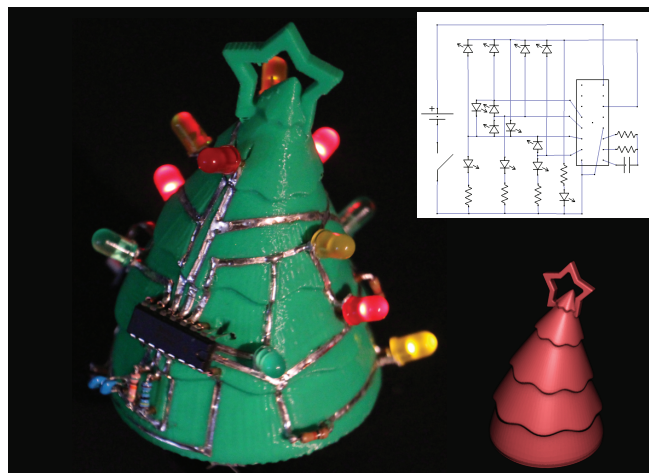


Figure 1. SurfCuit allows the user to design and fabricate surface mounted circuits on 3D prints (left). An illumination circuit (top right) is mounted on the surface of Christmas tree shape (bottom right).

the mechanical and electrical functions of simple objects. Our approach, which we call SurfCuit, allows the user to design and construct functional and durable electric circuits on the surface of 3D prints (see Fig. 1).

As a prototyping method, surface mounting has various advantages over embedded circuitry. First, construction is much easier since the parts are accessible from outside – the user does not need to insert the electric parts during printing, or try to fit components into tiny cavities. Second, it is easy to debug and repair surface-mounted circuits, while in many embedded-circuit applications this is difficult or impossible. Finally, the circuit design task is greatly simplified – compared to three-dimensional arrangements of cavities, fixtures, and wire channels, surface layouts are intuitive and efficient to create.

Our key challenges are (i) how to fabricate complex circuits on the surface of 3D prints and (ii) how to help the user perform the *circuit layout* task directly on the 3D surface in a computational design tool. Conductive inks do not readily adhere to 3D print plastics and their resistance is too high for many types of components. Instead, SurfCuit uses copper tape and tubes that are soldered together to achieve mechanically durable and highly conductive circuits. We leverage the fact that near soldering temperature, 3D-printed PLA plastic melts to a sticky viscous fluid that bonds well to copper material. To further enhance the fabrication process, our design tool adds shallow channels and holes to the 3D model before printing. These channels both help the user to re-create their virtual circuit traces in the physical world, and also help to

firmly affix the copper tape and through-hole parts onto the 3D print.

The traces which connect components in an electrical circuit must be physically isolated, i.e. they cannot intersect with each other. Because they exist in a (possibly curved) 2D space, with even a moderate number of traces it becomes very difficult to create the circuit without carefully planning out the traces ahead of time. Board layout and planning software like EAGLE is an essential tool for planar circuit design. However no existing circuit planning tool is applicable to arbitrary 3D surfaces. In addition, strips of copper tape cannot follow arbitrary paths on 3D surfaces, as many paths introduce too much torsion into the tape, which will result in kinks or tears. The strips should be laid out along *geodesics*, and without computational guidance it is very difficult to ensure that the 3D traces have this property. SurfCuit offers an interactive design tool that allows the user to easily adapt an existing planar circuit schematic to a 3D surface.

To demonstrate the capabilities of SurfCuit, we have designed and fabricated a variety of 3D objects with 3D circuitry. Some of these examples involve circuit complexity and levels of voltage/current that are well beyond what has been demonstrated in the literature. In addition, we also perform some destructive testing to demonstrate the robustness of our fabrication process. We show various examples of how our system facilitates the user's creation of functional 3D objects with electric circuits.

BACKGROUND AND RELATED WORK

Molded Interconnected Devices

Various advanced manufacturing technologies support the fabrication of circuitry that conforms to curved surfaces. For example, the Optomec Aerosol Jet system can create metal traces on simple 3D forms using laser metal deposition techniques. Another technology, Molded Interconnect Devices (MID), makes it possible to install circuitry on plastic surfaces. MID enables functional integration of electric circuit into small spaces, thus is often used for compact implementation of electronic products such as cell phones, cars, and advanced micro robots (see Figure 2). However, manufacturing such objects involves multi-axis laser engraving machines and etching/plating baths only found in advanced industrial facilities. Our SurfCuit system is inspired by these techniques, but our goal is to introduce MID-style fabrication in a more accessible context. In addition, currently there are no CAD tools for MID design. Our SurfCuit design tool is directly applicable to these advanced manufacturing methods.

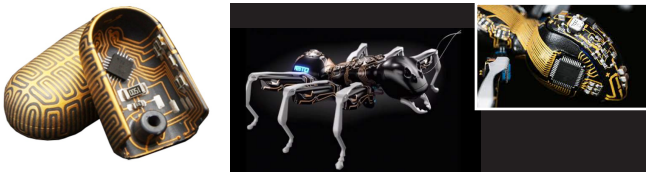


Figure 2. Example of molded interconnected devices. (Left) A robotic finger tip sensor by CITEC, Bielefeld University. (Right) FEST Bionic Ant Robot

Interactive 3D Prints

Various works in the human-computer interaction and fabrication literature have addressed the topic of adding interactivity to 3D prints. Techniques have been presented to convert 3D prints into sensors that include cameras [20], acoustics [15] and light-guides [25]. Sato et al. [19] studied frequency-dependent impedance properties to detect configurations of conductive 3D objects. Printput [7] and Capriate [23] use conductive filaments to convert the surface of 3D prints into capacitive touch sensors. However, such conductive filament traces have very high resistance, making it difficult to supply enough current to drive larger components (See Section Resistance Comparison).

Fabricating 3D circuit traces inside “tubes” passing through the interior of 3D prints was explored by Savage et al. [21]. This method is effective at hiding the circuitry, but also illustrates the challenge of repairing such devices. The capabilities of the automatic routing algorithms also limit the complexity of the circuits that can be designed with this approach. Hudson [9] studied the use of conductive threads for the yarn-based soft 3D printing. The commercially-available Voxel8 3D Printer [3] can embed circuits in a plastic print using conductive inks. The liquid ink can be deposited on the outer surface of prints, but only in upward-facing regions.

Prototyping Flat Circuits

Solderless prototyping methods such as traditional breadboards and LittleBits [5] are the fastest way to assemble a circuit, but these methods not suitable for permanent use. The resulting circuits are fragile, and also space consuming. Recently, advances in conductive ink have made it possible to directly deposit circuit traces on flat sheets using consumer ink-jet printers [11]. ShrinkCuit [13] uses Shrinky Dinks™ as the substrate to enhance complexity and conductivity of conductive-ink-based circuits. Sketch In Circuit [16] uses copper tape for prototyping traces on paper. Circuit Sticker [8] enables rapid prototyping of more complex circuits by pasting ready-made circuit boards on the top of traces. Ramakers et al. presented an interactive design system for circuits printed on paper [17]. The motivation behind these works is to enable creative circuit prototyping for “makers” and non-experts. We share this motivation, and our goal is to extend these ideas to arbitrary free-form 3D surfaces with a robust construction technique and interactive trace layout interface.

Circuit Design Tools

There are many circuit layout design tools for planar circuit boards. For example, commercial packages such as Eagle™, AutoTRAX™ and DipTrace™ provide comprehensive environments for design and simulation for PCB. Autodesk 123D Circuits [1] provides a layout design system for breadboards. Autodesk Project Wire provides 3D circuit layout for the Voxel8 printer [3]. Our SurfCuit 3D circuit design tool enables novices to quickly design 3D traces constrained to the surface of an arbitrary input mesh.

SURFCUIT CIRCUIT FABRICATION

Construction Procedure

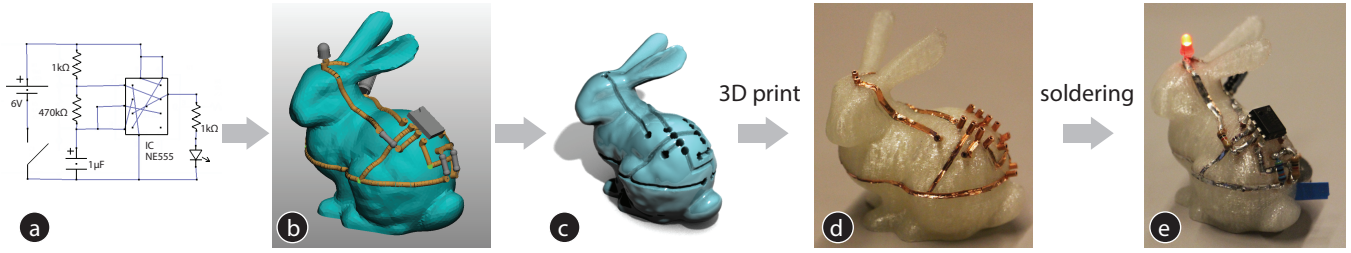


Figure 3. Workflow of SurfCuit. The user first draws a 2D schematic diagram of a circuit (a), then positions the electrical components on a 3D shape and connects them with curved traces (b). SurfCuit automatically generates channels and holes on the surface (c) to guide the user in placement of copper tapes and tubes. Finally, the user solders the copper pieces together to achieve a robust circuit on the 3D print.

The robustness of electrical connections is very important for permanent circuit construction — disconnection of a single trace can disable an entire circuit. However, constructing robust, highly conductive traces on a curved 3D-printed surfaces has been difficult. In this paper, we take advantage of the fact that PLA and ABS plastics that they melt into sticky, glue-like viscous fluids at around 200°C. Since the melting point of solder is also around that temperature, soldering on traces and pins melts the surrounding 3D plastic and strengthens the mechanical bonds between them. The result is that after cooling, the highly conductive copper traces are firmly affixed to the surface of the 3D print. Note that our approach is inspired by recent works that exploit melting behaviors for fabrication [14, 18]. However here we use melting for bonding, not for forming.

This melting technique creates strong connections, but actually using it on 3D surfaces requires some pre-planning. Computational design tools are necessary to plan the spatial layout for complex circuits. To guide the user in fabricating their design, we computationally generate channels and holes in the 3D surface before printing, which also helps to increase circuit robustness. Hence, the workflow of SurfCuit fabrication is as follows (see Fig. 3):

1. In the SurfCuit design tool, the user first draws a 2D circuit schematic diagram, then lays out the electric parts and traces on a 3D surface.
2. SurfCuit generates a 3D mesh with channels and holes that guide the user to install copper tape and tubes on a 3D print.
3. Then, the user covers these copper traces with solder, and solders the traces, pins, and electric parts together.
4. Finally, a thin layer of clear lacquer spray electrically protects the traces.

With SurfCuit, a novice maker can easily create complex functional 3D objects using widely-available single-material FDM printers. The soldering process requires exactly the same skills as fabricating a 2D circuit, which we already know that nearly anyone can learn to do. Applying solder to the copper tape is not difficult, as the solder naturally flows into the trace channels due to surface tension (see accompanying video). This soldering step also thickens the traces, further increasing mechanical robustness and electrical conductivity.

Construction Detail

Our circuit fabrication process is intended to be used with through-hole parts. Through-hole parts are desirable because they are widely available and are easy to manually solder. More importantly, through-hole parts achieve mechanically stronger bonds compared to surface mounted parts. Thus, they can be left exposed on the surface. The interval between pins is typically 1/10 inch = 2.54 mm for through-hole parts. Our technique maintains sufficient isolation between neighboring traces by keeping the trace width and pin diameter smaller than this interval (see Fig. 4).

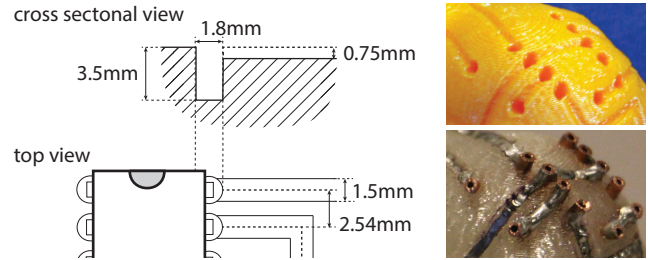


Figure 4. (left) Dimensions of holes and channels generated by our tool to guide copper tape and tube placement. We used 1.5 mm-wide copper tape and copper tubes with 1/16 inch (1.6 mm) diameter. The images on the right show the channels and holes in a 3D print (top), and the post-soldering result (bottom).

The channels and holes generated by our design tool are essential for the fabrication process. We arrived at this process after multiple iterations with simpler techniques. The benefits of this method include:

- The channels and holes help the user to accurately reproduce their complex virtual designs
- The channels and holes provide a large contact area between the copper traces and the 3D print.
- The holes help to temporarily hold components in place while they are soldered
- The channels partially enclose the traces and prevent them from peeling off.
- Since copper tape is placed in a V-shaped channel, it is easy to cover the channel with solder – surface tension makes the solder naturally flow into the channel.

The use of solid copper, rather than conductive inks or similar alternatives, is desirable for three reasons. First, copper has

comparatively high conductivity / low resistance (less than $0.5 \Omega/\text{m}$ in our traces). Thus, we are not limited by parasitic resistance when using the large electric currents typically necessary for driving small motors or incandescent lights. Secondly, its high heat conductivity makes it possible to bond copper to 3D-printed plastics via the application of heat. Finally, the copper is solderable. Solder quickly spreads on the surface of copper and creates mechanically and electrically robust bonding.

Compared to silver-based conductive inks (the most conductive), copper is also inexpensive and widely available. To make 1/16 inch traces, we split widely available 1/8 inch copper tape (about 10\$ for 50 m) in half using a rotary cutter. For copper pins, we used 1/16 inch copper tubes manufactured via K&S Engineering .Inc (three dollars for 1 m).

Resistance Comparison

During the development of our fabrication process, we experimented with various conductive inks and copper paints. However we found that most are difficult to apply to plastic surfaces. Liquid-based materials were also unreliable as a small crack results in electrical disconnection. This is especially problematic on the rough surfaces of 3D FDM prints, where the ink will tend to pool into the small cavities and channels produced by the printing process, creating highly variable thickness in the conductive layer. Furthermore, as reported in the previous studies, the electric resistance in the conductive liquids are very high (e.g. 11.48Ω for a $28\text{cm} \times 0.5\text{cm}$ trace using silver conductive ink [13] and $2 \Omega/\text{inch}$ for a 3mm diameter tunnel filled with copper paint [21]).

We also tried drawing traces with conductive PLA filament using a hand-held plastic extruder, however again the resistance is significantly higher than typical PCB traces, and thus cannot support many common circuits. For example, Black Magic 3D's Conductive Graphene Filament, which has one of the highest conductivity among the filaments on the market, still has $0.6 \Omega/\text{cm}^3$ volume resistivity. To achieve similar resistance as our traces ($0.5 \Omega/\text{m}$), the cross-sectional area should be at least $120 \text{ cm}^2 (=11\text{cm} \times 11\text{cm})$, vastly larger than the through-hole component pitch (2.54 mm). In other words, if we use conductive filament for traces, and the traces have a $1.5\text{mm} \times 1.5\text{mm}$ cross-section (small enough to connect to through-hole components), then a 20 cm long trace has more than $0.5 \text{ k}\Omega$ resistance. Such resistance causes over a 1V voltage drop with only 2mA current, which is barely enough to light an LED. Generally, conductive filaments are sufficient for capacitive touch sensors or blinking LEDs, however they fall short when attempting to drive common micro-controllers, actuators, and transducers.

Robustness Comparison

The user of copper tape in circuits-on-surfaces is not entirely novel, in particular copper tape is widely used in wearables and fashion tech. However these circuits are generally very fragile. To demonstrate the mechanical robustness of our approach, we made a qualitative comparison with a naïve method using destructive testing (see Figure 5).

In the naïve construction method, traces are just copper tapes placed on the 3D prints without the channels and solder. The tape generally has a sticky backing which is sufficient to hold it in place. The test circuits light a LED using a trace pattern that consists of more than twenty copper tapes (see Figure). We then brushed the circuits for one minute with nylon brushes to observe the robustness of the circuit. We first used a relatively soft nylon kitchen brush designed for washing dishes, and then switched to very hard nylon brush meant for scraping off rust.

The circuit with naïve construction immediately stopped functioning when the soft nylon brush was applied. This is because the connection between overlapping copper tape segments is particularly weak and breaks under small mechanical forces. After few additional seconds of brushing, the copper tape segments in the naïve construction start to peel off the plastic. After one minute, the copper traces in the naïve construction were severely damaged, to the point where the circuit would need to be entirely rebuilt. On the other hand, there was no visual damage to the SurfCuit traces even after the brushing with the hard brush. The circuit with SurfCuit construction stayed functional during the entire experiment. Please refer to the accompanying video for the detail of the comparison.

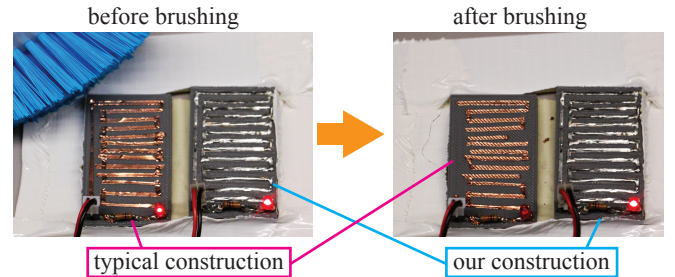


Figure 5. (left) LED lighting circuits with naïve construction and SurfCuit construction. (right) After brushing for one minute, the traces in the naïve construction were severely damaged while the SurfCuit traces were intact.

SURFCUIT DESIGN TOOL

Designing the trace layout for a complex circuit (e.g., 10+ connection points) typically requires planning before starting construction. Each trace that is added constrains the design space of future traces, because no traces can intersect. For the 2D flat circuits, we can plan on a paper or in vector-graphics CAD tools. However, since our circuits are on 3D free-form surfaces, we cannot lay out traces on a 2D flat geometry. It is also not practical to lay out traces as 3D space curves, as keeping them "on the surface" is very cumbersome. Hence, we developed a domain-specific interactive CAD tool which allows users to arrange parts and traces on arbitrary 3D surfaces.

User interface of SurfCuit design tool

Our SurfCuit design tool has two modes: 2D schematic design mode and 3D part and trace layout mode (see Fig. 6). The schematic mode lets the user specify electronic parts and their connections in the form of a 2D diagram, while the 3D layout mode lets the user arrange the parts and traces on the

3D model’s surface. The user can switch back and forth between these modes during circuit editing. While the user edits the circuit in one mode, a small window highlights the electric parts or traces currently being edited in the other mode, making 2D/3D correspondence easy to understand. Note that we are inspired by existing works showing highly abstracted schematic diagram while editing complex models [26, 12].

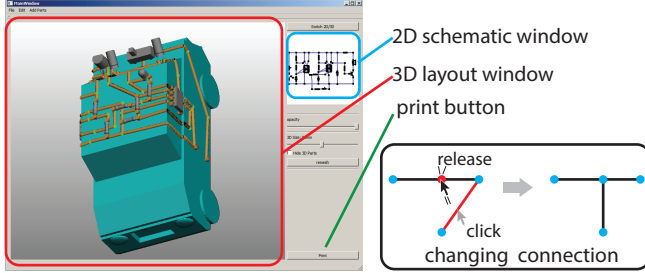


Figure 6. (left) SurfCuit 3D trace design interface. The user designs traces on a 3D object by dragging and rotating parts. The 2D schematic window is shown at the same time to facilitate understanding of the circuit structure. (right bottom) The user can change connectivity of traces by a simple gesture.

The schematic diagram is desirable as a circuit input rather than drawing traces directly on a 3D surface because schematics are symbolized and thus easily comprehensible. In our schematic editor, the user places symbols of electric parts and specifies these connections. To make the schematic tidy, the user also can add or delete a point on a trace and switch connectivity of points inside connected traces (see Fig. 6-right bottom). Note that thousands of schematic diagrams are widely available on the internet, for virtually any kind of circuit. Thus, inputting such diagrams does not require sophisticated electronics knowledge, a novice designer can simply copy an existing schematic.

The electric parts and their connections, specified in the 2D schematic window, are imported to the 3D layout window. The user lays out the parts by dragging and rotating them on the 3D surface. The traces are automatically generated on the surface in real-time during editing, making it easy for the user to lay out parts while avoiding intersecting traces. Similar to the schematic editor, the user can also add/delete points on the trace and change connectivity of points inside connected traces. Note that the 3D operations maintain the topological connection between pins of parts. These connections are specified in the 2D schematic window and are automatically reflected in the 3D layout window.

Finally, when the user presses the “print” button, the system creates the necessary channels and holes on the mesh that correspond to the design, and then exports the geometry for use in 3D printing software. Channels on the surface are generated using the stroke parametrization technique [22], which generates texture coordinates with minimum distortion around a stroke. We simply displace points of the mesh in the normal direction with respect to the distance computed from the parametrization.

Algorithmic detail of SurfCuit trace computation

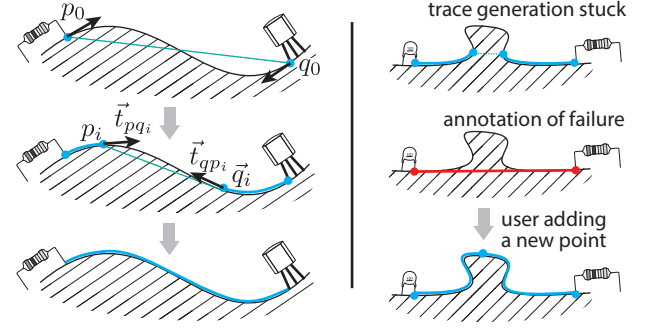


Figure 7. Surface trace generation algorithm. (left) Starting from two endpoints, the line segment between the two points is projected on the surface to find tangential directions for each point. We then incrementally slide each point along the surface in that projected direction, until they meet each other. (right) If the algorithm fails we annotate the failure to prompt the user to add an additional point.

SurfCuit updates the routing of traces interactively during the user’s editing. A trace is computed as a curve on the surface connecting two end points, each of which is either a pin of a user-specified trace midpoint. To make manual construction easy, traces should be as short and straight as possible, as this will introduce the smallest amount of twisting (torsion) in the copper tape. The shortest (and thus straightest) path on a surface connecting two points is called a *geodesic*.

There are various existing methods to compute geodesics (e.g., [24]), however true geodesics are expensive to compute. Thus, we use a heuristic method to estimate geodesics in real time (see Fig. 7-left). The basic idea behind our method is to start with the two endpoints p_0 and q_0 , and move them towards each other until they meet. The direction of movement is defined by a vector in the tangent plane at each point. We compute these directions by first finding a 3D direction, and then projecting into the tangent plane and normalizing. For a pair of points p_i and q_i , the tangent-plane directions are:

$$\vec{t}_{pq_i} = \frac{(\mathbf{I} - \vec{n}_{p_i} \otimes \vec{n}_{p_i})(\vec{q}_i - \vec{p}_i)}{|(\mathbf{I} - \vec{n}_{p_i} \otimes \vec{n}_{p_i})(\vec{q}_i - \vec{p}_i)|} \quad (1)$$

$$\vec{t}_{qp_i} = \frac{(\mathbf{I} - \vec{n}_{q_i} \otimes \vec{n}_{q_i})(\vec{p}_i - \vec{q}_i)}{|(\mathbf{I} - \vec{n}_{q_i} \otimes \vec{n}_{q_i})(\vec{p}_i - \vec{q}_i)|} \quad (2)$$

where \mathbf{I} is an identity matrix, and \vec{n}_p and \vec{n}_q are unit normal vectors at \vec{p} and \vec{q} . We then update the positions of points ($p_i \rightarrow p_{i+1}, q_i \rightarrow q_{i+1}$) by taking small steps in the directions of \vec{t}_{pq_i} and \vec{t}_{qp_i} , respectively. To keep the resulting point on the triangle mesh, we use discrete parallel transport [6]. First, a point on a triangle is updated in the direction of the triangle’s tangent vector \vec{t} until it hits the boundary (edge) of the triangle. Then, at the boundary we transform \vec{t} into the plane of the next triangle using a minimal rotation around the edge of the triangle, i.e. the rotation that takes the current triangle normal to the neighboring triangle normal. We stop this update when the p_i and q_i move 1 mm along the surface, and call them p_{i+1} and q_{i+1} .

This simple algorithm can fail when the normal of the surface (i.e., \vec{n}_p or \vec{n}_q) becomes parallel to $\vec{p}_i - \vec{q}_i$ (see Fig.7-

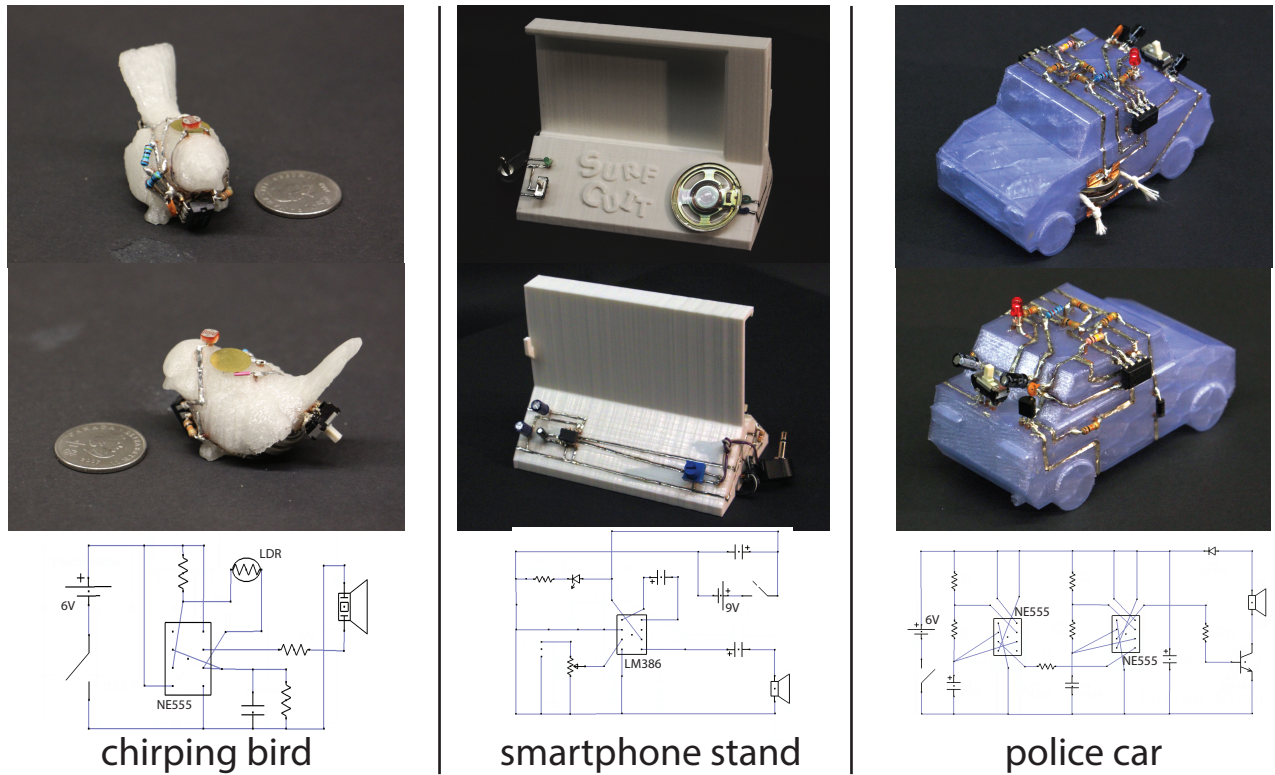


Figure 8. Fabricated SurfCuit examples and their schematic diagrams.

right). In such cases, we annotate failure as a line connecting p_0 and q_0 in order to prompt the user to add another point along the trace. Clearly this algorithm does not produce a bounded approximation to a geodesic, however it does produce exact geodesics for simple surface geometries such as planes or spheres. And in practice, we have found that it is highly effective at producing low-torsion 3D curves which are ideal for trace fabrication.

SURFCUIT EXAMPLES

To demonstrate the effectiveness of our approach, we present seven different examples created using our SurfCuit design tool and fabrication method. Each example is chosen to showcase various properties of SurfCuit. Specifically, we demonstrate integration of many different sensors (for light, touch, and sound), controller ICs, and transducers (for light, electromagnetic actuators, sound, and radio wave) into small spaces. Note that all the examples are fully self-contained and work without external controllers or power sources. Please refer to the accompanying video for more detail.

We can fabricate SurfCuits on a wide variety of input meshes. In these examples, the input meshes were taken from the Thingiverse™3D model repository (<http://www.thingiverse.com>). We did not need to specifically design new shapes from scratch to accommodate electric circuits.

Enriching 3D Prints with Sound and Lights

Christmas Tree

Figure 1 shows a Christmas tree (thing:608606) that blinks 13 LEDs in an asynchronized timing using a 16 pin timer IC, CD4060. This example has many components on a relatively small volume. 21 electric parts, 20 traces, and one 9-volt battery are integrated into the volume (12cm x 6cm x 6cm). This example also demonstrates the inclusion of an IC with many pins using SurfCuits. Such complex circuits typically need cables over the traces when constructed on a single-sided 2D board. However, in the SurfCuit, we can take advantage of the three dimensional structure of the object to avoid such cables. For example, if we cannot connect two parts on the front side without intersection, the trace can go around the back side. Because the circuit construction is three-dimensional, there are more degrees of freedom in the trace layout.

Chirping Birds

SurfCuit enables the fabrication of complex circuits in a very small volume. The chirping bird (Fig. 8) integrates a light theremin circuit, which uses a 555 timer IC and photoresistor, into a 3D bird shape (thing:359531). The light theremin circuit modulates the pitch of the sound according to the intensity of light received by an LDR sensor. Thus, a user can create chirping sounds by waving a hand on the top of the bird. This behavior significantly enhances the static bird geometry— not only does it generates sounds, it makes the bird a playable instrument. This example also demonstrates circuit integration into a small space. The part volume is very small (2.5cm x 3cm x 6cm) but because we use the full 3D space we can fit both the thermemin circuit and 2cm-diameter batteries. Our interactive layout tool allowed us to avoid ob-

structuring semantically-important features such as the bird's face, and place the batteries and switches in the occluded area behind the tail.

Smartphone Stand

The speaker-embedded iPhone stand (Fig. 8) augments a smartphone stand shape (thing:642881) by integrating a circuit using an LM386 timer IC to amplify the sound signal. This smartphone stand exemplifies the integration of geometric and electrical functionality. While the original geometry provides the function to hold a smartphone, the circuit amplifies the audio signal. 3D printing makes it easy to fabricate the precise shape needed. Achieving the same geometrical functionality is very difficult with flat circuits.

Police Car

The police car example (Fig. 8) integrates a circuit that blinks a LED beacon while a magnetic speaker generates siren sounds, which is modulated by two 555 timer ICs. Aside from the functionality of making sounds and lights of a police car, the surface mounted circuit also gives mechanical appearance to the 3D printed shape. The input mesh (thing:806770) is clearly a car but it lacks any detailed texture, in large part because the printer is limited to a single material. The circuitry on the car body gives the shape some definition, creating a more interesting machine-like appearance that would not be possible with 3D printing alone.

Dynamic 3D Prints with High-Current Circuits

As previously discussed, our traces have only small amounts of parasitic electric resistance, and thus can handle large amount of electric current (up to 1 Ampere, possibly more). This is enough current drive electro-magnetic actuators, allowing us to create mechanized objects.

Octopus Fan

Our Fan example (Fig. 9-left) demonstrates a touch-sensitive USB fan, where the user can toggle a DC motor fan on and off by touching specific locations of the print. The shape of the fan is designed to be clipped to the top of a computer monitor. Upon making physical contact, a 555 timer IC detects a small current transmitted through body, and toggles the motor control. We use a MOSFET to amplify the output and drive the 5V DC motor with about 1.0 A current. SurfCuit is convenient for fabricating touch-sensitive objects since the traces are naturally exposed to the surface. We successfully mounted long curved traces on the octopus's tentacle (thing:158069).

Cat Robot

In the waving cat example (Fig. 9-right), an ATtiny85, which is an Arduino-compatible programmable micro controller, drives two servo motors which wave the arm and shake the head of a cat statue (thing:163032). This robot also draw several hundreds milliampers of current at peak load. The use of a programmable micro-controller makes the interaction/behavior design of this object much more flexible. Again, SurfCuits highly conductive traces are critical to allowing the micro-controller to drive the various outputs.

Concealed Circuit

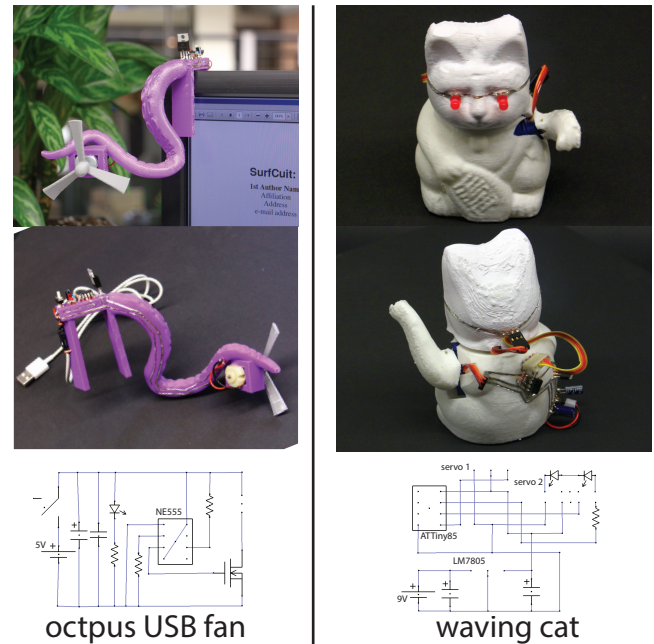


Figure 9. Examples of high-current electric circuits. (Left) Octopus USB fan with a touch switch. (Right) Cat robot waving its hand and head.

So far, our examples have placed the circuit components and trances on the exterior surface of objects. These circuits have increased the functionality and interactivity of the 3D prints via sensor-controlled transducers. However, adding on-surface circuits does involve modifying the original surface, which may be undesirable if the surface has specific functional or aesthetic purposes. An example of such a requirement arises in cases where we may wish to obscure the functionality of the circuit. Fig. 10 shows a FM transmitter that is concealed inside a shape of a squirrel (thing:11705). This squirrel spy could be used in covert recording or nanny-mic type applications. To create this object, we split the squirrel geometry along a curved partition surface, and then imple-

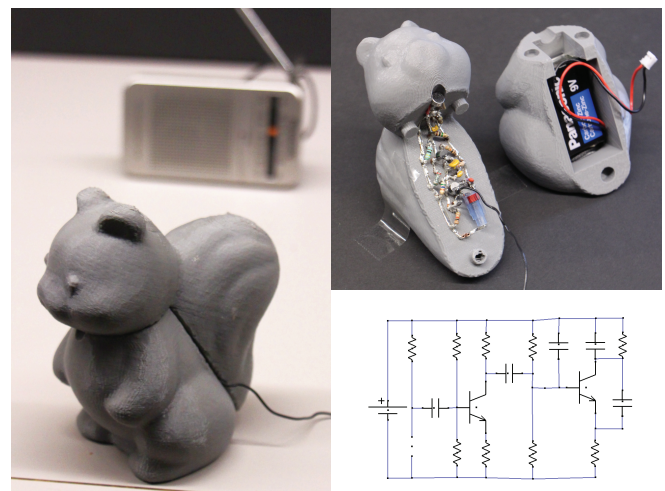


Figure 10. A covert Squirrel Spy which contains a concealed FM transmitter.

mented the circuit on the interior curved cross-sections. Using a curved partition, rather than a planar cut, allows us to design a parting line following concave regions on the surface, which are more easily concealed. We could of course create a larger cavity and installed a flat PCB board, but with SurfCuit we do not have to worry about the complexities of orienting and mounting the board. In this example, we also show that SurfCuit can handle very high-frequency circuits (about 100 MHz). Operating at such frequencies is very sensitive to the parasitic resistance that would be present with less robust fabrication processes.

Circuit as a Design Element

While in some cases we might wish to hide the circuit, in others we can actively use the circuit as part of the design aesthetic. Many people find beauty in circuits, as seen in the circuit jewelry (e.g., Circuit Breaker Labs [2]) and wearable fashion shows. In fact, many ground-breaking works of industrial design have integrated the internal engineering mechanisms into aesthetics of the design. Notable examples include Swiss watches showing the gear work or movement, the iMac G3's translucent body, and the intentionally-exposed functional and structural elements of the Pompidou museum in Paris. Although we cannot claim our own results as works of art, SurfCuit enables this aesthetic by making it easy for creative users to integrate circuitry elements into the external design of complex 3D shapes. For example, our featureless car above was turned into a police car with more interesting steampunk styling.

To further illustrate this concept, we created a circuit that illuminates EL (electro-luminescent) wires whose placement is designed based on an existing circuit-like facial tattoo (see Fig. 11). The core of the circuit is an inverter that converts 9V DC current to 120V AC current using 555 timer IC and a micro transformer. Although most of the traces do not contribute the function of the circuit, the texture of the metal traces completely changes the aesthetic of the otherwise smooth and monochrome 3D-printed head (thing:33503). This example also demonstrates the use of quite high-voltage circuits with SurfCuit, where insulation between traces is critical.

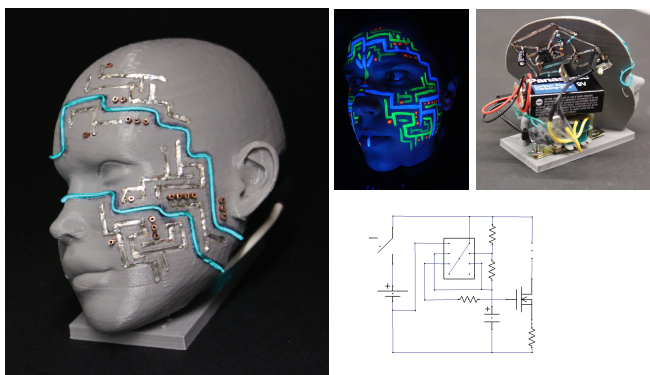


Figure 11. Facial tattoo model uses circuit's trace patterns as a design element (Left). This model is inspired by an artistic circuit tattoo by Faeriegem (Right).

FUTURE WORK

We are interested in making the circuit design system more intelligent by incorporating a circuit simulator (e.g., SPICE), a physics engine (e.g., Open Dynamics Engine) to simulate printed characters' dynamics, a schematic image recognition system [4], or an interactive sketch beautification system [10] to facilitate the user's creative circuit integrated 3D object design.

CONCLUSION

We presented SurfCuit: a system that integrates circuits into 3D prints by mounting them on the printed surface. Our construction method enables building rather complex, highly-conductive circuit patterns robustly on FDM-based 3D prints. Our interactive design system enables intuitive input and 3D layout of electric circuits on 3D geometry.

REFERENCES

- 123d circuits. <https://123d.circuits.io/>.
- Circuit breaker labs. <https://www.etsy.com/shop/CircuitBreakerLabs>.
- Voxel8: 3d electronics printing. <http://www.voxel8.co/>.
- Arvo, J., and Novins, K. Fluid sketches: Continuous recognition and morphing of simple hand-drawn shapes. In *Proceedings of the 13th Annual ACM Symposium on User Interface Software and Technology*, UIST '00, ACM (New York, NY, USA, 2000), 73–80.
- Bdeir, A. Electronics as material: Littlebits. In *Proceedings of the 3rd International Conference on Tangible and Embedded Interaction*, TEI '09, ACM (New York, NY, USA, 2009), 397–400.
- Bergou, M., Wardetzky, M., Robinson, S., Audoly, B., and Grinspun, E. Discrete elastic rods. *ACM Trans. Graph.* 27, 3 (Aug. 2008), 63:1–63:12.
- Burstyn, J., Fellion, N., Strohmeier, P., and Vertegaal, R. Printput: Resistive and capacitive input widgets for interactive 3d prints. In *Human-Computer Interaction INTERACT 2015*, J. Abascal, S. Barbosa, M. Fetter, T. Gross, P. Palanque, and M. Winckler, Eds., vol. 9296 of *Lecture Notes in Computer Science*. Springer International Publishing, 2015, 332–339.
- Hodges, S., Villar, N., Chen, N., Chugh, T., Qi, J., Nowacka, D., and Kawahara, Y. Circuit stickers: Peel-and-stick construction of interactive electronic prototypes. In *Proceedings of the 32nd Annual ACM Conference on Human Factors in Computing Systems*, CHI '14, ACM (New York, NY, USA, 2014), 1743–1746.
- Hudson, S. E. Printing teddy bears: A technique for 3d printing of soft interactive objects. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '14, ACM (New York, NY, USA, 2014), 459–468.

10. Igarashi, T., Matsuoka, S., Kawachiya, S., and Tanaka, H. Interactive beautification: A technique for rapid geometric design. In *Proceedings of the 10th Annual ACM Symposium on User Interface Software and Technology*, UIST '97, ACM (New York, NY, USA, 1997), 105–114.
11. Kawahara, Y., Hodges, S., Cook, B. S., Zhang, C., and Abowd, G. D. Instant inkjet circuits: Lab-based inkjet printing to support rapid prototyping of ubicomp devices. In *Proceedings of the 2013 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, UbiComp '13, ACM (New York, NY, USA, 2013), 363–372.
12. Kazi, R. H., Chevalier, F., Grossman, T., and Fitzmaurice, G. Kitty: Sketching dynamic and interactive illustrations. In *Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology*, UIST '14, ACM (New York, NY, USA, 2014), 395–405.
13. Lo, J., and Paulos, E. Shrinkycircuits: Sketching, shrinking, and formgiving for electronic circuits. In *Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology*, UIST '14, ACM (New York, NY, USA, 2014), 291–299.
14. Mueller, S., Kruck, B., and Baudisch, P. Laser origami: Laser-cutting 3d objects. *interactions* 21, 2 (Mar. 2014), 36–41.
15. Ono, M., Shizuki, B., and Tanaka, J. Touch & activate: Adding interactivity to existing objects using active acoustic sensing. In *Proceedings of the 26th Annual ACM Symposium on User Interface Software and Technology*, UIST '13, ACM (New York, NY, USA, 2013), 31–40.
16. Qi, J., and Buechley, L. Sketching in circuits: Designing and building electronics on paper. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '14, ACM (New York, NY, USA, 2014), 1713–1722.
17. Ramakers, R., Todi, K., and Luyten, K. Paperpulse: An integrated approach for embedding electronics in paper designs. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, CHI '15, ACM (New York, NY, USA, 2015), 2457–2466.
18. Sageman-Furnas, A. O., Umetani, N., and Schmidt, R. Melttables: Fabrication of complex 3d curves by melting. In *SIGGRAPH Asia 2015 Technical Briefs*, SA '15, ACM (New York, NY, USA, 2015), 14:1–14:4.
19. Sato, M., Poupyrev, I., and Harrison, C. Touché: Enhancing touch interaction on humans, screens, liquids, and everyday objects. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '12, ACM (New York, NY, USA, 2012), 483–492.
20. Savage, V., Chang, C., and Hartmann, B. Sauron: Embedded single-camera sensing of printed physical user interfaces. In *Proceedings of the 26th Annual ACM Symposium on User Interface Software and Technology*, UIST '13, ACM (New York, NY, USA, 2013), 447–456.
21. Savage, V., Schmidt, R., Grossman, T., Fitzmaurice, G., and Hartmann, B. A series of tubes: Adding interactivity to 3d prints using internal pipes. In *Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology*, UIST '14, ACM (New York, NY, USA, 2014), 3–12.
22. Schmidt, R. Stroke parameterization. *Comp. Graph. Forum* 32, 2 (2013).
23. Schmitz, M., Khalilbeigi, M., Balwierz, M., Lissermann, R., Mühlhäuser, M., and Steimle, J. Capricate: A fabrication pipeline to design and 3d print capacitive touch sensors for interactive objects. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology*, UIST '15, ACM (New York, NY, USA, 2015), 253–258.
24. Surazhsky, V., Surazhsky, T., Kirsanov, D., Gortler, S. J., and Hoppe, H. Fast exact and approximate geodesics on meshes. *ACM Trans. Graph.* 24, 3 (July 2005), 553–560.
25. Willis, K., Brockmeyer, E., Hudson, S., and Poupyrev, I. Printed optics: 3d printing of embedded optical elements for interactive devices. In *Proceedings of the 25th Annual ACM Symposium on User Interface Software and Technology*, UIST '12, ACM (New York, NY, USA, 2012), 589–598.
26. Zhu, B., Iwata, M., Haraguchi, R., Ashihara, T., Umetani, N., Igarashi, T., and Nakazawa, K. Sketch-based dynamic illustration of fluid systems. *ACM Trans. Graph.* 30, 6 (Dec. 2011), 134:1–134:8.